

# BEST AVAILABLE COPY

## REMARKS

Claims 1, 4-10, 12-16 and 19-22 were pending at the time of examination. Claims 1, 10, 16 and 21 have been amended. Claim 8 has been canceled. No new matter has been added. The applicants respectfully request reconsideration based on the foregoing amendments and these remarks.

### Interview summary

The applicants thank the Examiner for extending the courtesy of an interview on February 7, 2006. Claim 1 was discussed, in particular the step of determining a differential name and the fact that the claimed method uses information from the source program to compress an object file, such that a search can be obviated which is not the case in the cited art of record. No agreement was reached.

### Claim Rejections – 35 U.S.C. § 112

Claims 1, 10, 16 and 21 were rejected under 35 U.S.C 112, first paragraph as failing to comply with the written description requirement. In particular, the limitation recited as "...defines a context within which the differential name has an unambiguous meaning..." in claims 1, 10, 16, and 21 was objected to as not being mentioned in the specification. The applicants have amended the expression to read "... defines a context within which the differential name is distinct...", for which support can be found on page 12 of the specification, and which is in accordance with the Examiner's interpretation of the rejected limitation, as noted on page 3 of the most recent Office Action. The applicants respectfully submit that claims 1, 10, 16, and 21 now comply with the written description requirement, and that the rejection under 35 U.S.C 112, first paragraph be withdrawn.

### Claim Rejections – 35 U.S.C. § 103

Claims 1, 4-8, 10, 12, 14, 16, 19, 21 and 22 were rejected under 35 U.S.C § 103(a) as being unpatentable over U.S. Patent No. 5,991,173 to Unger et al. (hereinafter "Unger") in view U.S. Patent No. 6,163,811 to Porter et al., and in further view of Ainan "Storing text using integer codes," 1986, *Proceedings of the 11<sup>th</sup> conference on Computational linguistics* (hereinafter Ainan). The applicants respectfully traverse this rejection.

# BEST AVAILABLE COPY

Generally, the distinctions between the applicants' invention and the prior art can be described as follows: A source program is a description of an algorithm. An object program is an implementation of that algorithm, typically produced automatically with a compiler. The prior art describes various ways for compressing a source program, based on information in the source program. Similar techniques can be used for compressing an object program based on information in the object program. In contrast, the applicants' invention, as defined by the claims, pertains to methods and apparatus for compressing an object program based on structural information in the source program.

Turning now specifically to claim 1, claim 1 has been amended to more precisely identify the invention by incorporating the limitations of claim 8 (now canceled) and by further adding limitations specifying how the information in the source program is used in compressing the corresponding object code file. It is respectfully submitted that none of the cited prior art, alone or in combination, shows a compilation scheme where the end result is a compressed object code file containing differential names.

Claim 1 recites:

"receiving a source program written in a high-level programming language, the source program including one or more program symbols and non-program symbol information, wherein at least some of the one or more program symbols form containing scopes for other program symbols, the containing scopes being selected from the group consisting of: a namespace, a package, a module, a container object and a function;"

That is, the source program includes program symbols and non-program symbols. Some of the program symbols can form containing scopes for the other program symbols, such as namespaces, packages, modules, container objects, and functions. Thus, as explained in the specification on page 2, lines 6-22, the relationship between a program symbol name and another program symbol name defining its containing scope is known in the source program.

Next, claim 1, recites:

"encoding a program symbol name to produce an encoded program symbol name, without changing the non-program symbol information;"

That is, the program symbol names are encoded, that is, translated, to produce encoded program symbol names. This encoding occurs both for program symbol names that form containing scopes and program symbol names that do not form any containing scopes.

# BEST AVAILABLE COPY

Claim 1 further recites:

"when the encoded program symbol has a containing scope specified in the source program, determining a differential name for the encoded program symbol name relative to a base symbol identifying the containing scope for the program symbol as specified in the source program, wherein the containing scope defines a context within which the differential name is distinct, and wherein the differential name contains fewer characters than the encoded program symbol name and is formed at least in part by a sequence of characters constituting a subset of the encoded program symbol name;"

That is, a differential name is determined only for those encoded program symbol names that have a containing scope defined in the source program - not for any program symbols that are part of the source program. Expressed differently, the applicants' invention, as defined in claim 1, takes a translation of the original source program (i.e., the encoded program symbol names) and generates a compressed form (i.e., a differential name) of this translation. The encoded program symbols that are compressed in the applicants' invention never appear within the source program. In contrast, all the compression techniques in the cited art relate to compression of an original representation - which is analogous to the source program in the applicants' invention - and not on a translation of the original, as claimed.

Furthermore, all the compression techniques disclosed in the cited art all rely on the existence of a "corpus of previously encountered symbols" and a comparison or search of this corpus to generate the compressed version of the symbol. For example, Unger's technique (see col. 8, line 61) depends on identifying unique words in an unstructured source text and their frequency of occurrence, which is consistent with unstructured text compression; Porter's base/delta format is compression between a changed text and an original version of the same text; Aimon relies on a structured word list. In contrast, the applicants' invention does not require any such "corpus of previously encountered symbols." The differential name for the encoded program symbol name, as defined in claim 1, is defined relative to a base symbol that identifies the containing scope. As explained above, this containing scope is already defined in the source program. Thus, no search or lookup of a prior corpus is necessary.

Furthermore, claim 1 specifically identifies a limited number of containing scopes, such as, a namespace, a package, a module, a container object, and a function. No such contexts or containing scopes are mentioned in any of the cited prior art documents.

Claim 1 also specifies that the differential name "contains fewer characters than the encoded program symbol name and is formed at least in part by a sequence of characters

**BEST AVAILABLE COPY**

constituting a subset of the encoded program symbol name." Thus it is completely clear that the differential name is compressed with respect to the corresponding encoded name.

Finally, claim 1 recites:

"replacing the encoded program symbol name with the differential name to facilitate producing a compressed object code file including the differential name."

None of the cited compression techniques operate to generate a compressed object code file containing differential names. Unger compresses HTML objects "representing a stored picture, video, or sound" (see Unger, Fig. 8 and col. 8). These objects are clearly not the results of translating computer programs, as discussed above, and the end result is clearly not a compressed object code file containing differential names. Porter's base/delta format is compression between two versions of a changed text, whereas the applicants' invention pertains to generating a differential name for an encoded symbol relative to a containing scope. Porter does compression with the purpose of reducing the bandwidth of a source distribution system and makes no mention of compressing a stored form of the sources, let alone compressing a translated form of the sources, as required by claim 1. In both Unger and Porter, the compression occurs relative to an unstructured sequence of words. There is no semantic relationship between the "prior form" and the "new form." In contrast, the applicants' invention, as defined in claim 1, exploits a semantic relationship, which is defined in the source program. Ainon is directed to compression and processing of English text, and describes a family of word groups and an indication of a position within that family. In Ainon, the membership of a word within a word group does not contribute to compression, but only to analysis of the text, such as for confirming good grammar. In contrast, in the applicants' invention, the compression techniques uses on membership of an identifier within a scope when the compression is done. Furthermore, the word groups in Ainon are defined a priori based on the structure of the English language, whereas the applicants' invention relies on the structure of individual source programs for compression. For at least the above reasons, it is respectfully submitted that claim 1 is neither anticipated or rendered obvious by any of the cited prior art documents, alone or in combination, and it is respectfully submitted that the rejection of claim 1 be withdrawn.

Claim 16 is a *Beauregard* claim corresponding to claim 1. For reasons substantially similar to those set forth above, the applicants respectfully contend that the rejection of claim 16 is unsupported by the cited art and should be withdrawn.

# BEST AVAILABLE COPY

Claims 4-7 and 9 depend from claim 1, and the rejections of these claims are unsupported by the cited art for at least the reasons discussed above with regards to claim 1, and should be withdrawn.

Claims 19-20 depend from claim 16, and the rejections of these claims are unsupported by the cited art for at least the reasons discussed above with regards to claim 16, and should be withdrawn.

Claim 10 describes a method for generating encoded program symbol names in an uncompressed form, and was rejected for the same rationale that was set forth in the rejection of claim 1. Claim 10 contains limitations relating to program symbol names, base symbols, and differential program symbol names and formats. Consequently, for at least the reasons discussed above with regards to claim 1, the applicants respectfully contend that the rejection of claim 10 is unsupported by the cited art and should be withdrawn.

Claim 12 depends from claim 10, and the rejection of this claim is therefore unsupported by the cited art for at least the same reasons, and should be withdrawn.

Claim 21 is a *Beauregard* claim corresponding to claim 1. For reasons substantially similar to those set forth above, the applicants respectfully contend that the rejection of claim 16 is unsupported by the cited art and should be withdrawn.

Claim 13 was rejected for substantially the same reasons as claim 1. Claim 13 includes the limitation that the enhanced compiler includes "one or more differential names corresponding to the program symbol names." The program symbol names and the differential names have been discussed above with respect to the rejection of claim 1. For reasons substantially similar to those set forth above with regards to claim 1, the applicants respectfully contend that the rejection of claim 13 is unsupported by the cited art and should be withdrawn.

Claims 14 and 15 depend from claim 13, and the rejections of these claims are unsupported by the cited art for at least the reasons discussed with regards to claim 13, and should be withdrawn.

**BEST AVAILABLE COPY****Conclusion**

The applicants believe that all pending claims are allowable and respectfully request a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,  
BEYER WEAVER & THOMAS, LLP

*Fredrik Mollborn*  
Fredrik Mollborn  
Reg. No. 48,587

P.O. Box 70250  
Oakland, CA 94612-0250  
(650) 961-8300